AMENDMENTS

In the Claims

The following is a marked-up version of the claims with the language that is underlined ("___") being added and the language that contains strikethrough ("—") being deleted:

1. (Currently Amended) A method for preventing data entry via a data input screen on a client device, comprising:

rendering source code that defines ~~said~~ the data input screen in ~~said~~ the client device;

defining an executable script within ~~said~~ the source code; and

executing ~~said~~ the executable script in response to user input,

wherein ~~said~~ the executable script operates within ~~said~~ the client device to render ~~said~~ the data input screen inaccessible during processing of the user input to prevent duplicative execution of the executable script from subsequent user ~~input;~~ input, wherein upon completion of processing of the user input, the executable script renders the data input screen accessible;

wherein executing further comprises:

associating ~~said~~ the executable script with a predetermined z-index number for a web page; and

rendering inaccessible those data entry elements associated with ~~said~~ the web page that have a z-index number lower than ~~said~~ the predetermined z-index number.

2. (Currently Amended) The method as recited in claim 1, wherein ~~said~~ the source code comprises a tag-based language.

3. (Currently Amended) The method as recited in claim 2, wherein ~~said~~ the source code defines a membrane layer at a higher z-index level than other Web page elements, and

executing ~~said~~ the executable script further comprises changing a visibility attribute of ~~said~~ the membrane layer.

4. (Currently Amended) The method as recited in claim 1, wherein ~~said~~ the data input screen is received from a remote server and executing ~~said~~ the executable script is preformed solely on ~~said~~ the client device without any further processing by ~~said~~ the remote server.

5. (Currently Amended) An apparatus for preventing entries or submissions of data via an input screen displayed on a client device, comprising:

a central processing unit;

a memory:

a user input device;

a display; and

a browser adapted to render ~~said~~ the input screen on ~~said~~ the display,

wherein source code is provided to ~~said~~ the browser that contains instructions that are interpreted by ~~said~~ the browser to render ~~said~~ the input screen inaccessible after an executable script contained within source code is executed on ~~said~~ the client ~~device,~~ device to prevent duplicative execution of the executable script from subsequent user input, wherein the input screen is rendered accessible after execution of the executable script,

wherein ~~said~~ the source code further contains instructions which operate to:

generate association of ~~said~~ the executable script with a predetermined z-index number for a web page; and

render inaccessible those data entry elements associated with ~~said~~ the web page that have a z-index number lower than ~~said~~ the predetermined z-index number.

6. (Currently Amended) The apparatus as defined in claim 5, wherein ~~said~~ the executable

code is executed in response to user input.

7. (Currently Amended) The apparatus as defined in claim 5, wherein ~~said~~ the source code is a tag-based language.

8. (Currently Amended) The apparatus as defined in claim 5, wherein ~~said~~ the source code defines a membrane, and wherein a visibility attribute of ~~said~~ the membrane is changed by ~~said~~ the executable script.

9. (Currently Amended) The apparatus as defined in claim 8, wherein ~~said~~ the membrane is defined as layer in a cascading style sheet web page.

10. (Currently Amended) A computer-readable medium having computer-executable components comprising:

a form definition component defining a data input screen and a data submission field;

a style definition component defining a layer having a width and height at least as large as ~~said~~ the data submission field;

a function definition component responsive to ~~said~~ the data submission field, wherein upon execution of ~~said~~ the function definition component, ~~said~~ the layer operates to render ~~said~~ the data submission field inaccessible on ~~said form;~~ the form during execution of the function definition component, wherein the data submission field is rendered accessible upon completion of execution of the function definition component,

wherein ~~said~~ the computer-executable components are operable to perform the following:

associating ~~said~~ the executable script with a predetermined z-index number for a web page, and

rendering inaccessible those data entry elements associated with ~~said~~ the web page that have a z-index number lower than ~~said~~ the predetermined z-index number.

11. (Currently Amended) The computer-readable medium having computer-executable components as recited in claim 10, wherein ~~said~~ the layer is initially defined as hidden, and is made visible upon execution of ~~said~~ the function definition.

12. (Currently Amended) The computer-readable medium having computer-executable components as recited in claim 11, wherein ~~said~~ the layer comprises one of plural layers in a cascading style sheet web page.

13. (Currently Amended) The computer-readable medium having computer-executable components as recited in claim 10, wherein ~~said~~ the function definition component is executed in response to user operation of ~~said~~ the data submission field.

14. (Currently Amended) The computer-readable medium having computer-executable components as recited in claim 10 wherein ~~said~~ the function definition component is executed solely within a client device to prevent subsequent data entry via ~~said~~ the data input screen.

15. (Currently Amended) A method for preventing data entry to a server computer from a client computer, comprising:

receiving a request for an exchange of data from ~~said~~ the client computer;

defining an executable script within a source code, ~~said~~ the executable script operating in response to a client computer input and rendering a data input screen inaccessible to prevent duplicative processing of a subsequent input from ~~said~~ the client ~~computer;~~ computer during the

operation of the executable script, the input screen being rendered accessible in response to completion of the operation of the executable script; and

providing ~~said~~ the source code that defines ~~said~~ the data input screen;

wherein defining further comprises:

associating ~~said~~ the executable script with a predetermined z-index number for a web page; and

rendering inaccessible those data entry elements associated with ~~said~~ the web page that have a z-index number lower than ~~said~~ the predetermined z-index number.

16. (Currently Amended) The method as recited in claim 15, wherein ~~said~~ the source code comprises a tag-based language.

17. (Currently Amended) The method as recited in claim 16, wherein ~~said~~ the source code defines a membrane layer at a higher z-index number than other Web page elements, executing ~~said~~ the executable script further comprises changing a visibility attribute of ~~said~~ the membrane layer.

18. (Currently Amended) A method for preventing data entry to a web page comprising:

associating an executable script with ~~said~~ the web page;

permitting a first data input to ~~said~~ the web page;

executing, in response to ~~said~~ the first data input, ~~said~~ the executable script; and

preventing data entry to at least a portion of ~~said~~ the web page after execution of ~~said script,~~ the script to prevent duplicative processing of the first data input and a second data input, wherein preventing further comprises:

associating ~~said~~ the executable script with a predetermined z-index number for ~~said~~ the web page; and

rendering inaccessible those data entry elements associated with ~~said~~ the web page that have a z-index number lower than ~~said~~ the predetermined z-index ~~number.~~ number,

wherein upon completion of the execution of the script, the data entry elements associated with the web page are rendered accessible.

19. – 23. (Canceled)

24. (Currently Amended) A method for preventing data entry to a web page comprising:

associating an executable script with ~~said~~ the web page;

determining if ~~said~~ the web page used z-index numbers;

permitting a first data input to ~~said~~ the web page;

executing, in response to ~~said~~ the first data input, ~~said~~ the executable script; and

preventing data entry to at least a portion of ~~said~~ the web page after execution of ~~said script,~~ the script to prevent duplicative processing of the first data input and a second data input, wherein preventing further comprises:

associating ~~said~~ the executable script with a predetermined z-index number for ~~said~~ the web page if ~~said~~ the web page supports using ~~said~~ the z-index number;

associating ~~said~~ the executable script with a division of ~~said~~ the web page if ~~said~~ the web page does not support using ~~said~~ the z-index number;

rendering inaccessible those data entry elements associated with ~~said~~ the web page by rendering ~~said~~ the division of ~~said~~ the web page visible over ~~said~~ the data entry elements if ~~said~~ the web page does not support using ~~said~~ the z-index number; and

rendering inaccessible those data entry elements associated with ~~said~~ the web page that have a z-index number lower than ~~said~~ the predetermined z-index number if ~~said~~ the web page supports using ~~said~~ the z-index ~~number.~~ number,

wherein upon completion of the execution of the script, the data entry elements associated with the web page are rendered accessible.